Kernel Modified Quadratic Discriminant Function for Online Handwritten Chinese Characters Recognition

Duanduan Yang and Lianwen Jin
School of Electronics and Information, South China University of technology, Guangzhou, 510640, P.R.China
ddyang@scut.edu.cn , eelwjin@scut.edu.cn

Abstract

The Modified Quadratic Discriminant Function has been used successfully in handwriting recognition, which can be seen as a dot-product method by eigen-decomposition of the covariance matrix. Therefore, it is possible to expand MQDF to high dimension space by kernel trick. This paper presents a new kernel-based method, Kernel Modified Quadratic Discriminant Function (KMQDF) for online Chinese Characters Recognition. Experimental results show that the performance of MQDF is improved by the kernel approach.

1. Introduction

Statistical techniques have been widely used in Chinese character recognition problems. The quadratic discriminant function (QDF) is one of them, which is based on the assumption of multivariate Gaussian density for each class under the Bayes decision theory framework. The modified QDF (MQDF) proposed by Kimura *et al.* [1] aims to improve the computation efficiency and classification performance of QDF via eigenvalue smoothing, which have been used with great success for handwriting recognition [eg. 2,3]. The difference of MQDF from the QDF is that the eigenvalues of minor axes are set to a constant. The motivation behind this is to smooth the parameters for compensating for the estimation error on finite sample size.

Recently, the kernel-based learning machines, e.g., support vector machines (SVM)[4], kernel principal component analysis (KPCA)[5], and kernel Fisher discriminant analysis (KFD)[6], have received a lot of interest in the fields of statistical pattern recognition and machine learning. The basic idea of kernel methods is finding a mapping ϕ such that, in new space, problem solving is easier (e.g. linear). But the mapping is left implicit. The *kernel* represents the similarity between two objects defined as the dot-product in this new vector space. Thus, the kernel methods can be easily generalized to dot-products-based pattern recognition algorithms. QDF and MQDF can be also regarded as dot-product methods by eigen-decomposition of the covariance matrix. It is expected that the performance of MQDF can be improved to solve some complex problems when it is generalized to a new high-dimension space by kernel trick. In this paper, we present a new kernel method, KMODF, for online handwritten Chinese character recognition. Experiments show that KMQDF with proper kernel function outperforms MQDF.

2. A Brief Introduction of MQDF

Let us represent a pattern with a feature vector x, a posteriori probability is computed by Bayes rule:

$$P(\omega_i \mid x) = p(x \mid \omega_i)P(\omega_i) / p(x) \tag{1}$$

where $P(\omega_i)$ is the a priori probability of class ω_i, $p(x \mid \omega_i)$ is the class probability density function and $p(x)$ is the mixture density function. Since $p(x)$ is independent of class label, the nominator of (1) can be used as the discriminant function for classification:

$$g(\omega_i \mid x) = p(x \mid \omega_i)P(\omega_i) \tag{2}$$

The Bayesian classifier is reduced to QDF under the Gaussian density assumption with varying restrictions. Assume the probability density function of each class is multivariate Gaussian:

$$p(x \mid \omega_i) = \frac{1}{2\pi^{d/2}\left|\Sigma\right|_i^{1/2}} \exp[\frac{(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)}{2}] \tag{3}$$

where μ_i and Σ_i denote the mean vector and the covariance matrix of class ω_i, respectively. Inserting

Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)
0-7695-2822-8/07 $25.00 © 2007 IEEE

(3) into (2), taking the negative logarithm and omitting the common terms under equal priori probabilities, the QDF is obtained as:

$$g(x,\omega_i)=(x-\mu_i)^T\,\Sigma_i^{-1}(x-\mu_i)+\log|\Sigma_i| \tag{4}$$

The QDF is actually a distance metric in the sense that the class of minimum distance is assigned to the input pattern. By eigen-decompostion, the covariance matrix can be diagonalized as:

$$\Sigma_i = \mathrm{B}_i\Lambda_i\mathrm{B}_i^T \tag{5}$$

where Λ_i is a diagonal matrix formed by the eigenvalues of Σ_i, B_i is formed by the corresponding eigenvectors.

According to (5), the QDF can be rewritten in the form of eigenvectors and eigenvalues:

$$\begin{aligned} g(x,\omega_i) &= [\mathrm{B}_i^T(x-\mu_i)]^T\,\Lambda_i^{-1}\mathrm{B}_i^T(x-\mu_i)+\log|\Lambda_i| \\ &= \sum_{j=1}^{d}(1/\lambda_{ij})[\beta_{ij}^T(x-\mu_i)]^2+\sum_{j=1}^{d}\log(\lambda_{ij}) \end{aligned} \tag{6}$$

Replacing the minor eigenvalues with a constant δ_i, the modified quadratic discriminant function (MQDF) [1] is obtained as follows:

$$\begin{aligned} g_2(x,\omega_i) &= \sum_{j=1}^{k}(1/\lambda_{ij})[\beta_{ij}^T(x-\mu_i)]^2+\sum_{j=1}^{k}\log(\lambda_{ij}) \\ &\quad+\sum_{j=k+1}^{d}(1/\delta_i)[\beta_{ij}^T(x-\mu_i)]^2+(d-k)\log(\delta_i) \\ &= \sum_{j=1}^{k}(1/\lambda_{ij})[\beta_{ij}^T(x-\mu_i)]^2+\sum_{j=1}^{k}\log(\lambda_{ij}) \\ &\quad+(1/\delta_i)r_i(x)+(d-k)\log(\delta_i) \end{aligned} \tag{7}$$

where k denotes the number of principal axed and $r_i(x)$ is the residual of subspace projection:

$$r_i(x)=\|x-\mu_i\|^2-\sum_{j=1}^{k}\,[\beta_{ij}^T(x-\mu_i)]^2 \tag{8}$$

The parameter δ_i of MQDF can be set to a constant as the following equation [2]:

$$\delta_i=(tr(\Sigma_i)-\sum_{j=1}^{k}\lambda_{ij})/(d-k)=\sum_{j=k+1}^{d}\lambda_{ij}\,/(d-k) \tag{9}$$

where $tr(\Sigma_i)$ denotes the trace of covariance matrix.

The advantage of MQDF is multifold. First, it overcomes the bias of minor eigenvalues such that the classification performance can be improved. Second, for computing the MQDF, only the principle eigenvectors and eigenvalues are to be stored so that the memory space is reduced. Third, the computation effort is largely saved because the projections to minor axes are not computed [2].

3. Kernel MQDF

As a statistical algorithm, MQDF can detect stable patterns robustly and efficiently from a finite data samples. By embedding the data samples in a suitable feature space by kernel trick, it is possible that MQDF can perform better than in the original feature space. According to this idea, we subsequently present a new kernel-based method, KMQDF algorithm, in this section.

For a given nonlinear mapping function Φ, the input data space IR^n can be mapped into the feature space H. As a result, a pattern in the original input space IR^n is mapped into a potentially much higher dimensional feature vector in the feature space H. Since the feature space H is possibly infinite dimensional and the orthogonality needs to be characterized in such a space, it is reasonable to view H as a Hilbert space. An initial motivation of KMQDF is to perform MQDF in the feature space H. However, it is difficult to do so directly because it is computationally very intensive to compute the dot products in a high-dimensional feature space. Fortunately, kernel techniques can be introduced to avoid this difficulty. The algorithm can be actually implemented in the input space by virtue of kernel tricks. The explicit mapping process is not required at all.

Given a set of M training samples x $(x_{i1},x_{i2},\ldots x_{iM})$ in IR^n, labeled with the ith class, the covariance operator on the feature space H can be constructed by

$$\Sigma_i^{\Phi}=(1/M)\sum_{j=1}^{M}(\Phi(x_{ij})-m_i^{\Phi})(\Phi(x_{ij})-m_i^{\Phi})^T \tag{10}$$

where $m_i^{\Phi}=(1/M)\sum_{j=1}^{M}\Phi(x_{ij})$. In a finite-dimensional Hilbert space, this operator is generally called covariance matrix. Since every eigenvalue of a positive operator is nonnegative in a Hilbert space, it follows that all nonzero eigenvalues of Σ_i^{Φ} are positive. It is the positive eigenvalues that are of interest to us.

It is easy to show that every eigenvector of Σ_i^{Φ}, β, can be linearly expanded by

$$\beta=\sum_{j=1}^{M}a_j\Phi(x_{ij}) \tag{11}$$

To obtain the expansion coefficients, let us denote $Q_i=[\Phi(x_{i1})\ldots\ldots\Phi(x_{iM})]$, and form an M*M Gram matrix $\tilde{R}_i=Q_i^TQ_i$, whose elements can be determined by virtue of kernel tricks:

$$\tilde{R}_{i(u,v)}=\Phi(x_{iu})^T\Phi(x_{iv})=(\Phi(x_{iu})\bullet\Phi(x_{iv}))=\ker(x_{iu}\bullet x_{iv}) \tag{12}$$

We can centralize $\tilde{R}_i$ by $R_i=\tilde{R}_i-1_M\tilde{R}_i-\tilde{R}_i1_M+1_M\tilde{R}_i1_M$, where $1_M=(1/M)_{M\times M}$.

On the other hand, we can denote Σ_i^{Φ} and R_i in terms of Q_i as flowing:

$$\Sigma_i^\Phi = (1/M)(Q_i - Q_i 1_M)(Q_i - Q_i 1_M)^T \tag{13}$$

$$R_i = (Q_i - Q_i 1_M)^T (Q_i - Q_i 1_M) \tag{14}$$

Considering an eignevector-eignevalue pair γ_i and λ_i of R_i, we have:

$$(1/M)(Q_i - Q_i 1_M)(Q_i - Q_i 1_M)^T (Q_i - Q_i 1_M)\gamma_i \ \ldots = (1/M)\lambda_i (Q_i - Q_i 1_M)\gamma_i \tag{15}$$

From (14) and (15), we can get:

$$\Sigma_i^\Phi (Q_i - Q_i 1_M)\gamma_i = (\lambda_i / M)(Q_i - Q_i 1_M)\gamma_i \tag{16}$$

Equation (16) implies that $(Q_i - Q_i 1_M)\gamma_i, \lambda_i / M$ is an eigenvetor-eigenvalue pair of Σ_i^Φ. Furthermore, the norm of $(Q_i - Q_i 1_M)\gamma_i$ is given by:

$$\|(Q_i - Q_i 1_M)\gamma_i\|^2 = \gamma_i^T (Q_i - Q_i 1_M)^T (Q_i - Q_i 1_M)\gamma_i = \lambda_i \tag{17}$$

Therefore the corresponding normalized eigenvetor of Σ_i^Φ is $\beta_i = (Q_i - Q_i 1_M)\gamma_i / \sqrt{\lambda_i}$.

Calculate the orthonormal eigenvetors $\gamma_{i1}, \gamma_{i2} \cdots \gamma_{im}$ of R_i corresponding to the m largest positive eigenvalues, $\lambda_{i1} \ge \lambda_{i2}, \ldots \ge \lambda_{im}$. The orthonormal eigenvetors $\beta_{i1}, \beta_{i2} \ldots \beta_{im}$ of Σ_i^Φ corresponding to the m largest positive eignevalues, $\lambda_{i1}/M, \lambda_{i2}/M, \ldots \lambda_{im}/M$, which are $\beta_{ij} = (Q_i - Q_i 1_M)\gamma_{ij} / \sqrt{\lambda_{ij}}$, $j = 1,2,3..m$.

Analogizing equation (7), in new feature space ,we have the KMQDF:

$$\begin{aligned}
g_2^\Phi(x, \omega_i) &= \sum_{j=1}^{k} (1/\lambda_{ij}^\Phi)[\beta_{ij}^{\Phi^T} (\Phi(x) - m_i^\Phi)]^2 + \sum_{j=1}^{k} \log(\lambda_{ij}^\Phi) \\
&+ \sum_{j=k+1}^{d} (1/\delta_i^\Phi)[\beta_{ij}^{\Phi^T} (\Phi(x) - m_i^\Phi)]^2 + (d-k)\log(\delta_i^\Phi) \\
&= \sum_{j=1}^{k} (M/\lambda_{ij}) \{[(Q_i - Q_i 1_M)\gamma_{ij} / \sqrt{\lambda_{ij}}]^T [\Phi(x) - Q_i 1_{M_1}]\}^2 \\
&+ \sum_{j=k+1}^{d} (1/\delta_i^\Phi) \{[(Q_i - Q_i 1_M)\gamma_{ij} / \sqrt{\lambda_{ij}}]^T [\Phi(x) - Q_i 1_{M_1}]\}^2 \\
&+ \sum_{j=1}^{k} \log(\lambda_{ij}/M) + (d-k)\log(\delta_i^\Phi) \\
&= \sum_{j=1}^{k} \frac{M}{\lambda_{ij}^2} [\gamma_{ij}^T (R_{it} - 1_M R_{it} - \tilde{R}_i 1_{M_1} + 1_M \tilde{R}_i 1_{M_1})]^2 \\
&+ \sum_{j=k+1}^{d} \frac{1}{\delta_i^\Phi \lambda_{ij}} [\gamma_{ij}^T (R_{it} - 1_M R_{it} - \tilde{R}_i 1_{M_1} + 1_M \tilde{R}_i 1_{M_1})]^2 \\
&+ \sum_{j=1}^{k} \log(\lambda_{ij}/M) + (d-k)\log(\delta_i^\Phi)
\end{aligned} \tag{18}$$

where $R_{it} = [(\Phi(x_{i1}) \bullet \Phi(x)), (\Phi(x_{i2}) \bullet \Phi(x)), \ldots (\Phi(x_{iM}) \bullet \Phi(x))]^T$ $1_{M\_1} = (1/M)_{M \times 1}$ and $\delta_i = \sum_{j=k+1}^{d} (\lambda_{ij}/M)/(d-k)$

We can utilizes the invariance of Euclidean distance to simply equation (18)

$$\begin{aligned}
g_2^\Phi(\omega_i \mid x) &= \sum_{j=1}^{k} \frac{M}{\lambda_{ij}^2} [\gamma_{ij}^T (R_{it} - 1_M R_{it} - \tilde{R}_i 1_{M_1} + 1_M \tilde{R}_i 1_{M_1})]^2 \\
&+ \sum_{j=1}^{k} \log(\lambda_{ij}/M) + \frac{1}{\delta_i^\Phi} r_i^\Phi(x) + (d-k)\log(\delta_i^\Phi)
\end{aligned} \tag{19}$$

where
$$\begin{aligned}
r_i^\Phi(x) &= \|(\Phi(x) - m_i^\Phi)\|^2 - \sum_{j=1}^{k} [\beta_{ij}^{\Phi^T} (\Phi(x) - m_i^\Phi)]^2 \\
&= (\Phi(x) \bullet \Phi(x)) - 2 * (1_{M_1})^T \bullet R_{it} + (1_{M_1})^T \tilde{R}_i (1_{M_1}) \\
&- \sum_{j=1}^{k} \frac{1}{\lambda_{ij}} [\gamma_{ij}^T (R_{it} - 1_M R_{it} - \tilde{R}_i 1_{M_1} + 1_M \tilde{R}_i 1_{M_1})]^2
\end{aligned} \tag{20}$$

It is expected that the KMQDF algorithm can embed the data in a suitable feature space, in which we can use MQDF algorithm to discover pattern more easily.

5. Feature Extraction

Directional features have been widely used for Chinese character recognition with great success. The 8-directional features have been demonstrated to be very effectiveness for online Chinese character recognition in [7]. Extraction of 8-directional features mainly consists of the following four steps:

Step 1. Preprocessing: Given an online handwritten character sample, a series of processing, including linear size normalization, adding imaginary strokes, nonlinear shape normalization, equidistance re-sampling, and smoothing, are performed to derive a 64×64 normalized online character sample.

Step 2. Extraction of online 8-directiona features: The 8-directional feature vector is derived according the direction projections along the character strokes.

Step 3. Mapping to 8 directional pattern images. The 8-directional features from all online points of a character sample is directly mapped to eight 2-D directional image patterns according to the feature values of the corresponding points.

Step 4. Extraction of blurred directional features: Each directional pattern image is first divided uniformly into 8 × 8 grids whose centers are treated as spatial sampling points. Then a Gaussian filter is applied at each sampling point to generate the final blurred directional feature.

Details of the 8-directional feature extraction can be found in [7].

6. Experiments

60 sets of online handwritten Chinese characters (each set consists of 3755 categories of GB2312-80 level 1 Chinese characters) were used in our experiments. The characters were written by 60 different individuals. And all the characters are written naturally with no constraint in stroke order, stroke number and writing style. Fig. 1 shows some samples.

We use 50 sets for training, the rest characters are used as testing samples.

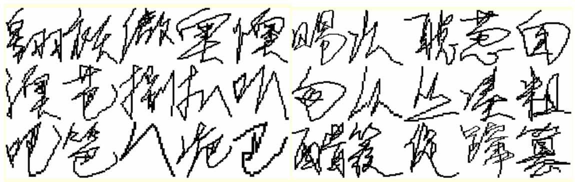

Figure 1. Some samples for our experiments

First, we demonstrate the performance of different polynomial kernels (see table 1) using the same number of dominant eigenvertors($k=10$), it can be seen that $\ker(x \bullet y)=(x \bullet y)^{0.2}$ has the best performance.

Table 1 the KMQDF performance using different polynomial kernels with the same number of dominant eigenvertors ($k=10$)

Polynomial kernel	$\ker(x \bullet y) = (x \bullet y)^{0.2}$	$\ker(x \bullet y) = (x \bullet y)^{0.8}$	$\ker(x \bullet y) = (x \bullet y)^{1.4}$	$\ker(x \bullet y) = (x \bullet y)^{2}$
Recognition rate(%)	94.11	93.72	93.64	92.02

Next, we show the KMQDF performance using RBF kernel with the different numbers of dominant eigenvertors. From table 2, it is shown the classifier with $k=10$ outperform others.

Table 2 the KMQDF performance using RBF($\ker(x \bullet y)=\exp(|x-y|^2/\sigma^2)$) with different k

Number of dominant eigenvertors k	10	20	30	40
Recognition rate(%)	93.01	92.83	92.12	91.67

At last, we compare the performances of KMQDF and MQDF with different k**.**

Table 3 the performance comparison of KMQDF($\ker(x \bullet y)=(x \bullet y)^{0.2}$) and MQDF with different k

Number of dominant eigenvertors k	10	20	30	40
KMQDF recognition rate(%)	94.11	93.85	93.01	92.51
MQDF recognition rate(%)	93.15	92.86	92.10	91.73

From table3, we can see that the highest recognition rate, 94.11%, is obtained by KMQDF. Comparing with that of MQDF, 93.15%, it is shown that the recognition performance of MQDF is improved by almost 1%. Due to the great memory and computation required, currently we only use 50 sets of characters to train KMQDF. If more samples are involved in the training, it is possible to improve the recognition rate furthermore. In addition, it is worth to mention that the selection of proper kernel function is very important to guarantee the performance of KMQDF.

7. Conclusion

As a new attempt of using kernel approach for online Chinese character recognition, this paper presents a new kernel-based algorithm, Kernel MQDF, which performs MQDF in a potentially much higher dimensional feature space. We give the detail derivation of the KMQDF algorithm. Experimental results show that the proposed KMQDF can outperform MQDF classifier.

8. Acknowledgments

This paper is sponsored by the following research foundations: New Century Excellent Talent Program of MOE of China (No.NCET-05-0736).

9. References

[1] F.Kimura, K.Takashina, S.Tsuruoka, and Y.Miyake, "Modified quadratic discriminant functions and the application to Chinese character recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.9, No.1, pp.149-153,1987.

[2] Cheng-Lin Liu, Sako H, and Fujisawa H, "Discriminative Learning Quadratic Discriminant Function for Handwriting Recognition", *IEEE Transaction on Neural networks,* Vol.15, No.2, pp.430-444, 2004.

[3] Hailong Liu, Xiaoqing Ding, "Handwritten character recognition using gradient feature and quadratic classifier with multiple discrimination schemes", In of Proceedings of the Eight International Conference on Document Analysis and Recognition, Seoul, Korea, August. Vol. 1 pp.19-23,2005

[4] V.Vapnik, *The nature of stastical leaning Theory*, New York, Springer, 1995.

[5] B. Scholkopf, A. Smola, and K.R. Muller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem", *Neural Computation*, Vol. 10, No. 5, pp. 1299-1319, 1998.

[6] Jian Yang, Alejandro F. Frangi, Jing-yu Yang, David Zhang, "KPCA Plus LDA: A Complete Kernel Fisher Discriminant Framework for Feature Extraction and

Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 27, No. 2, pp.230-244, 2005.

[7] Zhen-Long BAI and Qiang HUO, "A Study On the Use of 8-Directional Features For Online Handwritten Chinese Character Recognition", In of Proceedings of the Eight International Conference on Document Analysis and Recognition, Seoul, Korea, August. Vol. 1 pp.232-236,2005.

Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)
0-7695-2822-8/07 $25.00